Incremental Mining from News Streams

Seokkyung Chung
University of Southern California, USA

Jongeun Jun
University of Southern California, USA

Dennis McLeod
University of Southern California, USA

INTRODUCTION

With the rapid growth of the World Wide Web, Internet users are now experiencing overwhelming quantities of online information. Since manually analyzing the data becomes nearly impossible, the analysis would be performed by automatic data mining techniques to fulfill users' information needs quickly.

On most Web pages, vast amounts of useful knowledge are embedded into text. Given such large sizes of text collection, mining tools, which organize the text datasets into structured knowledge, would enhance efficient document access. This facilitates information search and, at the same time, provides an efficient framework for document repository management as the number of documents becomes extremely huge.

Given that the Web has become a vehicle for the distribution of information, many news organizations are providing newswire services through the Internet. Given this popularity of the Web news services, text mining on news datasets has received significant attentions during the past few years. In particular, as several hundred news stories are published everyday at a single Web news site, triggering the whole mining process whenever a document is added to the database is computationally impractical. Therefore, efficient incremental text mining tools need to be developed.

BACKGROUND

The simplest document access method within Web news services is keyword-based retrieval. Although this method seems effective, there exist at least three serious drawbacks. First, if a user chooses irrelevant keywords, then retrieval accuracy will be degraded. Second, since keyword-based retrieval relies on the syntactic properties of information (e.g., keyword counting), *semantic gap* cannot be overcome (Grosky, Sreenath, & Fotouhi, 2002). Third, only expected information can be retrieved since the specified keywords are generated from users' knowledge space. Thus, if users are unaware of the airplane crash that occurred yesterday, then they cannot issue a query about that accident even though they might be interested.

The first two drawbacks stated above have been addressed by query expansion based on domain-independent ontologies. However, it is well known that this approach leads to a degradation of precision. That is, given that the words introduced by term expansion may have more than one meaning, using additional terms can improve recall, but decrease precision. Exploiting a manually developed ontology with a controlled vocabulary would be helpful in this situation (Khan, McLeod, & Hovy, 2004). However, although ontology-authoring tools have been developed in the past decades, manually constructing ontologies whenever new domains are encountered is an error-prone and time-consuming process. Therefore, integration of knowledge acquisition with data mining, which is referred to as *ontology learning*, becomes a must (Maedche & Staab, 2001).

To facilitate information navigation and search on a news database, clustering can be utilized. Since a collection of documents is easy to skim if similar articles are grouped together, if the news articles are hierarchically classified according to their topics, then a query can be formulated while a user navigates a cluster hierarchy. Moreover, clustering can be used to identify and deal with near-duplicate articles. That is, when news feeds repeat stories with minor changes from hour to hour, presenting only the most recent articles is probably sufficient. In particular, a sophisticated incremental hierarchical document clustering algorithm can be effectively used to address high rate of document update. Moreover, in order to achieve rich semantic information retrieval, an ontology-based approach would be provided. However, one of the main problems with concept-based ontologies is that topically related concepts and terms are not explicitly linked. That is, there is no relation between *court-attorney*, *kidnap-police*, and etcetera. Thus, concept-based

ontologies have a limitation in supporting a topical search. In sum, it is essential to develop incremental text mining methods for intelligent news information presentation.

MAIN THRUST

In the following, we will explore text mining approaches that are relevant for news streams data.

Requirements of Document Clustering in News Streams

Data we are considering are high dimensional, large in size, noisy, and a continuous stream of documents. Many previously proposed document clustering algorithms did not perform well on this dataset due to a variety of reasons. In the following, we define application-dependent (in terms of news streams) constraints that the clustering algorithm must satisfy.

1. **Ability to determine input parameters:** Many clustering algorithms require a user to provide input parameters (e.g., the number of clusters), which is difficult to be determined in advance, in particular when we are dealing with incremental datasets. Thus, we expect the clustering algorithm not to need such kind of knowledge.
2. **Scalability with large number of documents:** The number of documents to be processed is extremely large. In general, the problem of clustering *n* objects into *k* clusters is NP-hard. Successful clustering algorithms should be scalable with the number of documents.
3. **Ability to discover clusters with different shapes and sizes:** The shape of document cluster can be of arbitrary shapes; hence we cannot assume the shape of document cluster (e.g., hyper-sphere in *k*-means). In addition, the sizes of clusters can be of arbitrary numbers, thus clustering algorithms should identify the clusters with wide variance in size.
4. **Outliers Identification:** In news streams, outliers have a significant importance. For instance, a unique document in a news stream may imply a new technology or event that has not been mentioned in previous articles. Thus, forming a singleton cluster for the outlier is important.
5. **Efficient incremental clustering:** Given different ordering of a same dataset, many incremental clustering algorithms produce different clusters, which is an unreliable phenomenon. Thus, the incremental clustering should be robust to the input sequence. Moreover, due to the frequent document insertion into the database, whenever a new document is inserted it should perform a fast update of the existing cluster structure.
6. **Meaningful theme of clusters:** We expect each cluster to reflect a meaningful theme. We define "meaningful theme" in terms of precision and recall. That is, if a cluster (*C*) is about "Turkey earthquake,'' then all documents about "Turkey earthquake'' should belong to *C*, and documents that do not talk about "Turkey earthquake" should not belong to *C*.
7. **Interpretability of resulting clusters:** A clustering structure needs to be tied up with a succinct summary of each cluster. Consequently, clustering results should be easily comprehensible by users.

Previous Document Clustering Approaches

The most widely used document clustering algorithms fall into two categories: partition-based clustering and hierarchical clustering. In the following, we provide a concise overview for each of them, and discuss why these approaches fail to address the requirements discussed above.

Partition-based clustering decomposes a collection of documents, which is optimal with respect to some predefined function (Duda, Hart, & Stork, 2001; Liu, Gong, Xu, & Zhu, 2002). Typical methods in this category include center-based clustering, Gaussian Mixture Model, and etcetera. Center-based algorithms identify the clusters by partitioning the entire dataset into a pre-determined number of clusters (e.g., *k*-means clustering). Although the center-based clustering algorithms have been widely used in document clustering, there exist at least five serious drawbacks. First, in many center-based clustering algorithms, the number of clusters needs to be determined beforehand. Second, the algorithm is sensitive to an initial seed selection. Third, it can model only a spherical (*k*-means) or ellipsoidal (*k*-medoid) shape of clusters. Furthermore, it is sensitive to outliers since a small amount of outliers can substantially influence the mean value. Note that capturing an outlier document and forming a singleton cluster is important. Finally, due to the nature of an iterative scheme in producing clustering results, it is not relevant for incremental datasets.

Hierarchical (agglomerative) clustering (HAC) identifies the clusters by initially assigning each document to its own cluster and then repeatedly merging pairs of clusters until a certain stopping condition is met (Zhao & Karypis, 2002). Consequently, its result is in the form of a tree, which is referred to as a *dendrogram*. A dendrogram is represented as a tree with numeric levels associated to its branches. The main advantage of HAC lies in its ability